

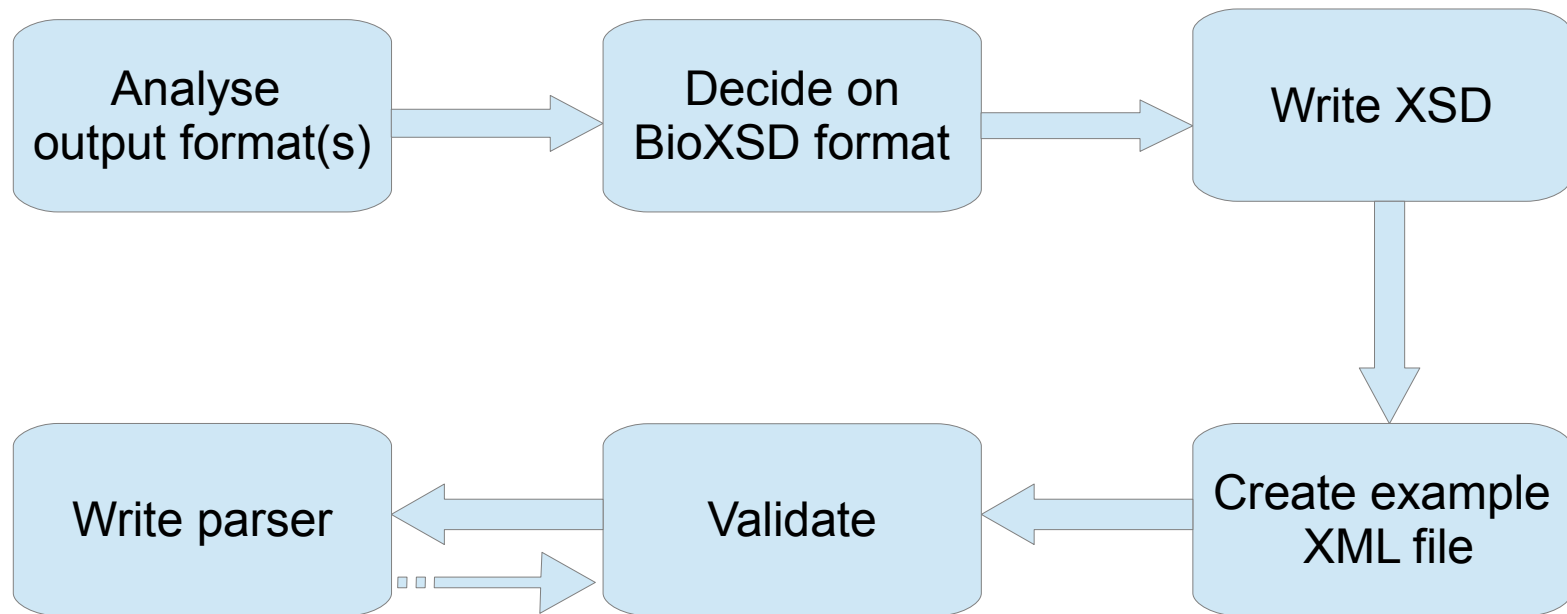
Bioinformatics Lab SS 2013

Semester Challenge - XML, XSD, BioXSD and Ontologies

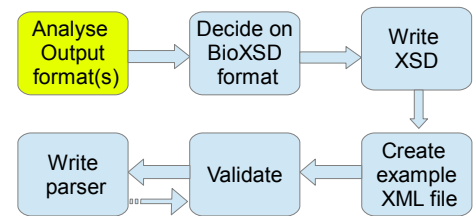
Sonja Ansorge

9. 7. 2013

Workflow



Preparations



NAME

ncoils - prediction of coiled-coil secondary structure elements

SYNOPSIS

```
ncoils [OPTION] < [FASTA FILE]
ncoils -f < /usr/share/doc/ncoils/1srya.fa
```

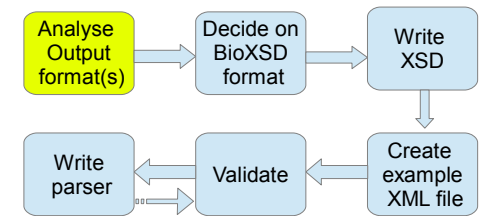
DESCRIPTION

ncoils is a program that compares a sequence to a database of known parallel two-stranded coiled-coils and derives a similarity score. By comparing this score to the distribution of scores in globular and coiled-coil proteins, the program then calculates the probability that the sequence will adopt a coiled-coil conformation.

OPTIONS

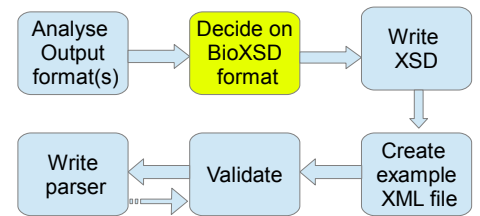
```
-f, -fasta
    fasta output - coils as 'x', like '-x' in seg
-c concise mode - which sequences have any coils (and how many)
-min_seg <int>
    for concise mode - only report sequence if >= min coil segments
-min_P <float>
    minimum P to define coil segment; DEFAULT = 0.5
-win <int>
    window size; DEFAULT = 21
-w weight heptad positions a&d the same as b,c,e,f,g
-v verbose/debug mode - print extra junk
```


Ncoils Output



```
>1srya SERYL-TRNA SYNTHETASE (E.C.6.1.1.11) (THERMUS THERMOPHILUS, STRAIN HB27)
MVDLKRLRQEPEVFHRAIREKGVAlDLEAxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxRNQVAKRVPK
APPEEKEALIARGKALGEExxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxVPLPPWPGAPVGGEeANREI
KRVGGPPEFSFPPLDHVALMEKNGWWEPRISQVSGSRSYALKGDLALYELALLRFAMDFM
ARRGFLPMTLPSYAREKAFLGTGHFPAYRDQVWAlAETDLYLTGTAEVVLNALHSGEILP
YEALPLRYAGYAPAFRSEAGSFGKDVRGLMRVHQFHKVEQYVLTEASLEASDRAFQELLE
NAEEILRLLELPYRLVEVATGDMGPGKWRQVDIEVYLPSEGRYRETHSCSALLDWQARRA
NLRYRDPEGRVRYAYTLNNTALATPRILAMLLENHQLQDGRVRVPQALIPYMGKEVLEPC
```


BioXSD Elements



biomolecular **sequences**

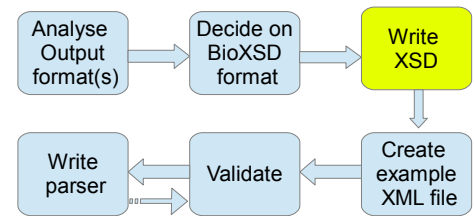
sequence **alignments**

sequence and genome **features** (annotation)

references to data, identifiers/accessions, ...

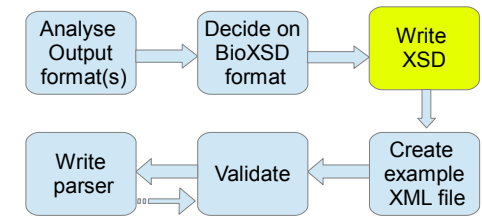
See slides by Matúš Kalaš

XSD



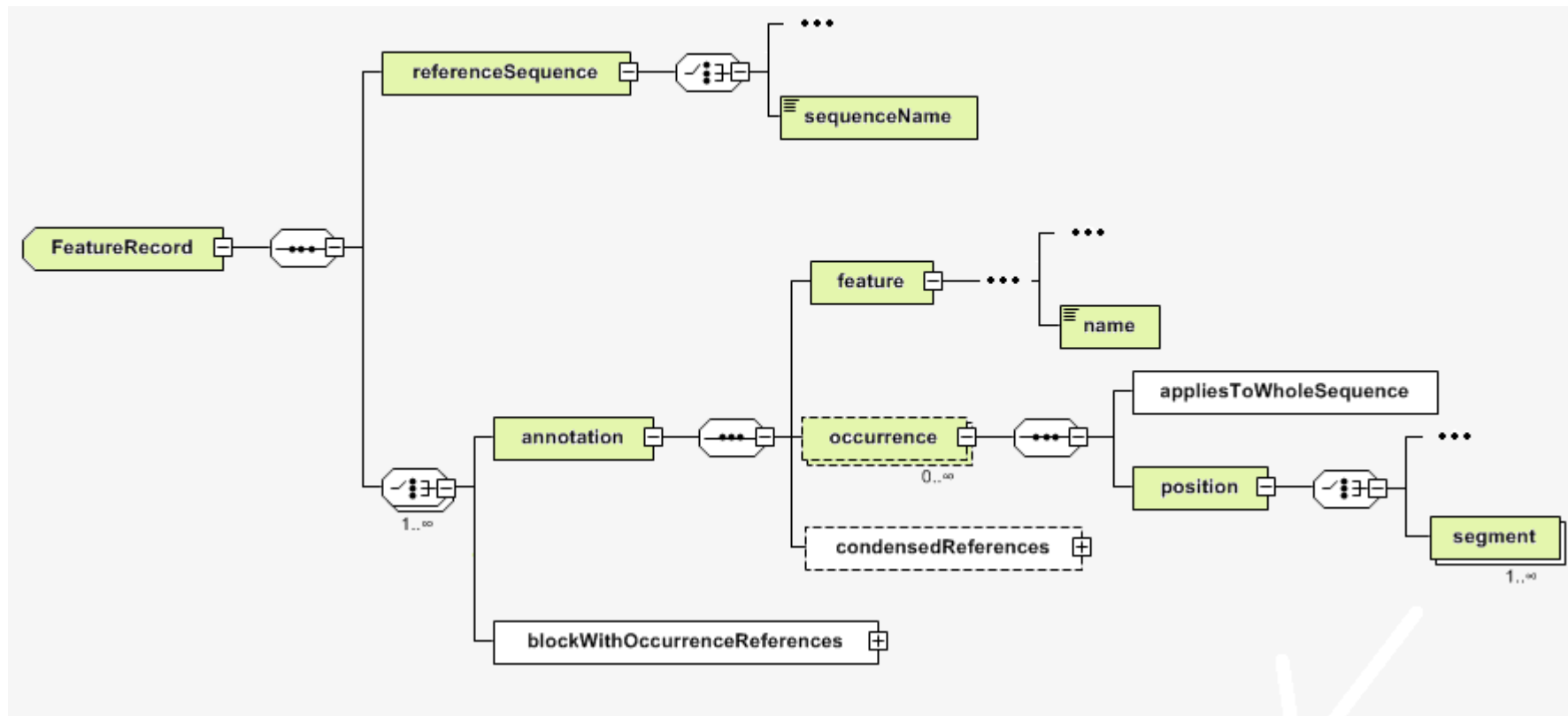
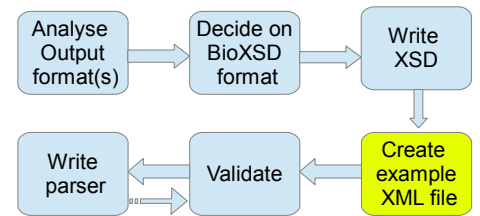
```
1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <xs:schema
3     xmlns="http://i12r-tbl.informatik.tu-muenchen.de/~verena"
4     targetNamespace="http://i12r-tbl.informatik.tu-muenchen.de/~verena"
5     xmlns:bx="http://bioxsd.org/BioXSD-1.1"
6     xmlns:xs="http://www.w3.org/2001/XMLSchema"
7     xmlns:sawSDL="http://www.w3.org/ns/sawSDL"
8     attributeFormDefault="unqualified"
9     elementFormDefault="qualified"
10    version="1.0.0">
11    <xs:import namespace="http://bioxsd.org/BioXSD-1.1"
12              schemaLocation="http://bioxsd.org/BioXSD-1.1.xsd"/>
13    <xs:element name="ncoils" type="bx:FeatureRecord"
14              sawSDL:modelReference="http://edamontology.org/data_0877">
15      <xs:annotation>
16        <xs:documentation>The program predicts the coiled coil secondary
17                          structure predictions from protein sequences...
18        </xs:documentation>
19      </xs:annotation>
20    </xs:element>
21 </xs:schema>
```


EDAM Ontology

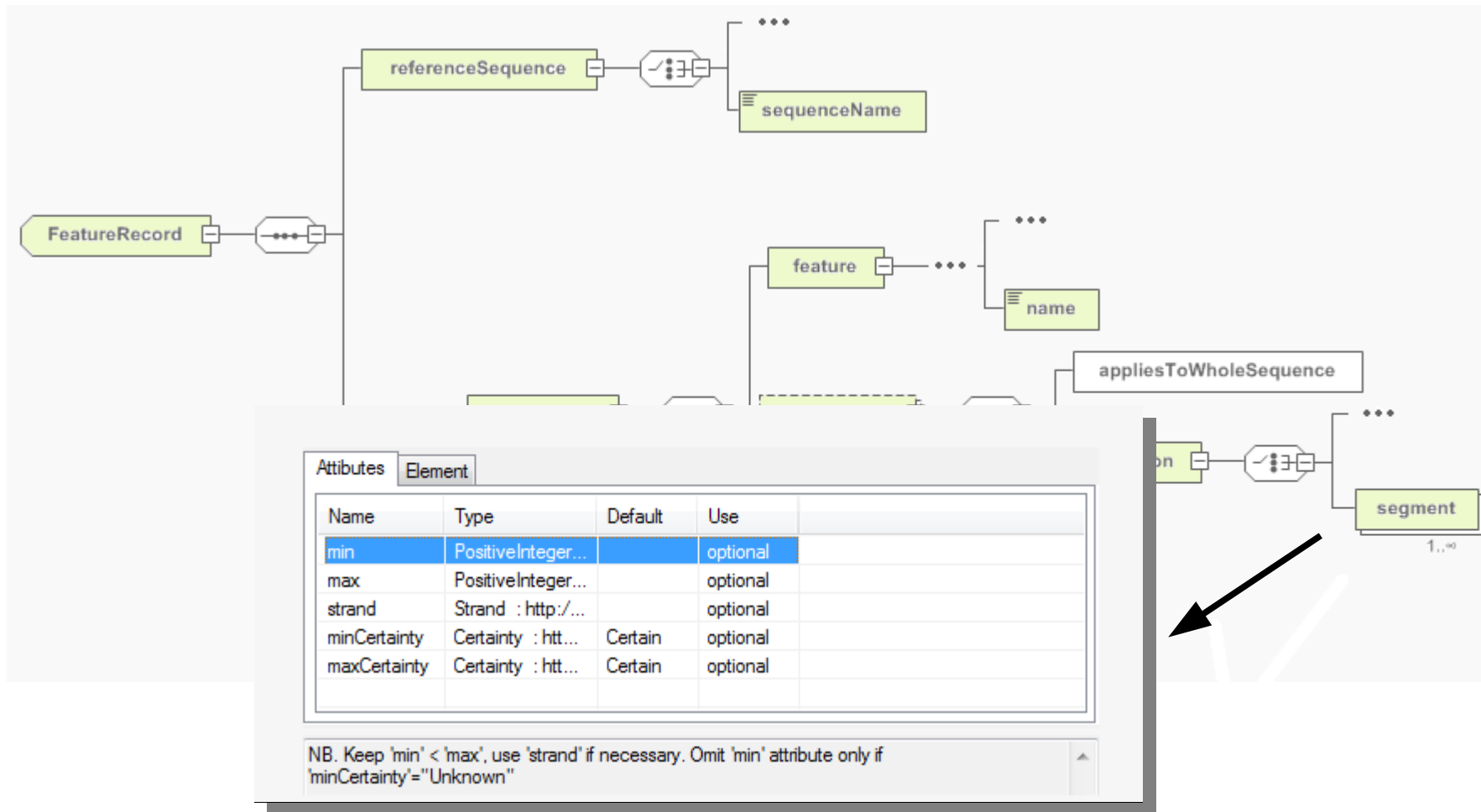
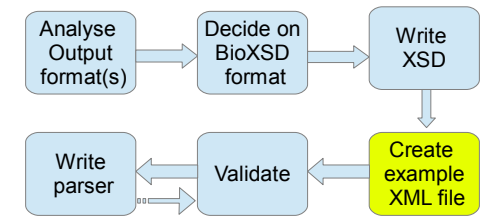


Preferred Name (<i>rdfs:label</i>)	Protein features (<u>super-secondary</u>)
Synonyms (<i>oboInOwl:hasExactSynonym</i>)	Protein structure report (super-secondary)
Definitions (<i>oboInOwl:hasDefinition</i>)	A report of predicted or actual super-secondary structure of protein sequence(s).
ID	data_0877
Full Id	http://edamontology.org/data_0877
comment	Super-secondary structures include leucine zippers, coiled coils, Helix-Turn-Helix etc.
Created in	beta12orEarlier
hasDefinition	A report of predicted or actual super-secondary structure of protein sequence(s).
hasExactSynonym	Protein structure report (super-secondary)
inSubset	data bioinformatics edam
namespace	data
label	Protein features (super-secondary)
subClassOf	<u>Protein structure report</u> <u>Protein features</u>

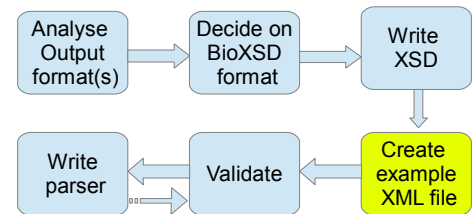
XSD Diagram



XSD Diagram

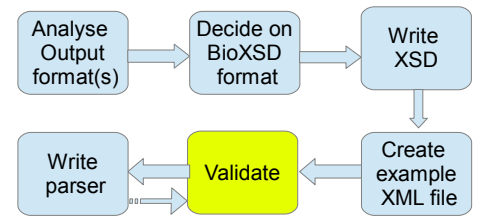


XML



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <ncoils xmlns="http://i12r-tbl.informatik.tu-muenchen.de/~verena"
3     xmlns:bx="http://bioxsd.org/BioXSD-1.1"
4     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5     xsi:schemaLocation="http://i12r-tbl.informatik.tu-muenchen.de/~verena
6     http://i12r-tbl.informatik.tu-muenchen.de/~verena/ncoils.xsd">
7     <bx:referenceSequence>
8         <bx:sequenceName>1srya_SERYL-TRNA_SYNTHETASE_(E.C.6.1.1.11)
9         |_____(THERMUS_THERMOPHILUS,_STRAIN_HB27)|
10    </bx:sequenceName>
11 </bx:referenceSequence>
12 <bx:annotation>
13     <bx:feature>
14         <bx:name>coiled coils secondary structure</bx:name>
15     </bx:feature>
16     <bx:occurrence>
17         <bx:position>
18             <bx:segment min="29" max="50"/>
19             <bx:segment min="79" max="100"/>
20         </bx:position>
21     </bx:occurrence>
22 </bx:annotation>
23 </ncoils>
```


Validation



- `apt-get install libxerces-c-samples`
- `StdInParse -n -s -f -v=always < ncoils.xml`
- `=> stdin: 330 ms (10 elems, 12 attrs, 77 spaces, 111 chars)`
- <http://www.utilities-online.info/xsdvalidation/>

XML

```
<?xml version="1.0" encoding="utf-8"?>
<ncoils xmlns="http://i12r-tbl.informatik.tu-muench
  xmlns:bx="http://bioxsd.org/BioXSD-1.1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://i12r-tbl.informatik.tu-muench
    http://i12r-tbl.informatik.tu-muench/BioXSD-1.1/BioXSD-1.1.xsd">
  <bx:referenceSequence>
    <bx:sequenceName>lsrya_SERYL-TRNA_SYNTHETASE_(E
  </bx:referenceSequence>
  <bx:annotation>
    <bx:feature>
      <bx:name>coiled coils secondary structure</bx:
    </bx:feature>
    <bx:occurrence>
      <bx:position>
        <bx:segment min="29" max="50"/>
        <bx:segment min="79" max="100"/>
      </bx:position>
    </bx:occurrence>
  </bx:annotation>
```

Check XML Well Formed

XSD Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no">
<xs:schema
  xmlns="http://i12r-tbl.informatik.tu-muench
  targetNamespace="http://i12r-tbl.informatik
  xmlns:bx="http://bioxsd.org/BioXSD-1.1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:sawSDL="http://www.w3.org/ns/sawSDL"
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  version="1.0.0">
  <xs:import namespace="http://bioxsd.org/Bio
    schemaLocation="http://i12r-tbl.informatik
  <xs:element name="ncoils" type="bx:FeatureF
    sawSDL:modelReference="http://i12r-tbl.informatik
    <xs:annotation>
      <xs:documentation>The program
    </xs:documentation>
    </xs:annotation>
  </xs:element>
```

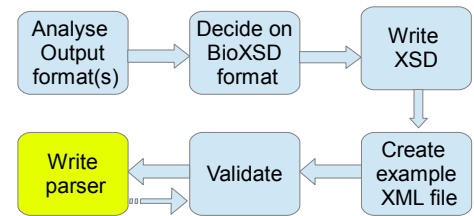
Check XSD Validity

Validate XML against XSD

Result

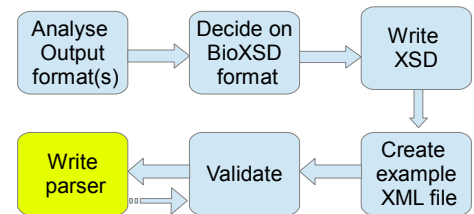
The XML is Well Formed and Valid.

Parser with Perl and LibXML



```
1 use XML::LibXML;
2 #parse input file |
3 # ...
4 #create new xml document
5 my $document = XML::LibXML::Document->new( '1.0', 'utf-8' );
6 # create the ROOT element
7 my $root = $document->createElement ( 'ncoils' );
8 #add ATTRIBUTES to the root (namespace etc.)
9 $root->addChild ( $document->createAttribute ( 'xmlns' => 'http://i12r-tbl.informatik.tu-muenchen.de/~verena' ) );
10 $root->addChild ( $document->createAttribute ( 'xmlns:bx' => 'http://bioxsd.org/BioXSD-1.1' ) );
11 $root->addChild ( $document->createAttribute ( 'xmlns:xsi' => 'http://www.w3.org/2001/XMLSchema-instance' ) );
12 $root->addChild ( $document->createAttribute ( 'xsi:schemaLocation' => 'http://i12r-tbl.informatik.tu-muenchen.de/~verena
13                                     http://i12r-tbl.informatik.tu-muenchen.de/~verena/ncoils.xsd' ) );
14 #insert CHILD elements here:
15 #...
16 # completes the settings of the root/document
17 $document->setDocumentElement ( $root );
18 my $xml = $document->toString('1');
19 print $xml;
```


Parser – Child Elements



```
1 # create a new child ELEMENT/TAG for the referenceSequence
2 my $refSeqXML = $document->createElement('bx:referenceSequence');
3 my $headerXML = $document->createElement('bx:sequenceName');
4 $headerXML->addChild($document->createTextNode($header));
5 $refSeqXML->addChild($headerXML);
6 $root->addChild($refSeqXML);
7
8 # create a new child ELEMENT/TAG for the annotation
9 my $annoXML = $document->createElement('bx:annotation');
10 # feature TAG
11 my $featureXML=$document->createElement('bx:feature');
12 my $nameXML=$document->createElement('bx:name');
13 $nameXML->addChild($document->createTextNode('coiled coils secondary structure'));
14 $featureXML->addChild($nameXML);
15 $annoXML->addChild($featureXML);
16 # occurrence TAG
17 my $occXML = $document->createElement('bx:occurrence');
18 my $posXML = $document->createElement('bx:position');
19 # coiled coils positions parsed from sequence
20 getCoils($sequence);
21 # and converted into segment TAGs
22 for (my $i=0; $i<scalar(@start_pos); $i++ ){
23     my $segXML = $document->createElement('bx:segment');
24     $segXML->addChild($document->createAttribute(min => $start_pos[$i]));
25     $segXML->addChild($document->createAttribute(max => $end_pos[$i]));
26     $posXML->addChild($segXML);
27 }
28 $occXML->addChild($posXML);
29 $annoXML->addChild($occXML);
30 $root->addChild($annoXML);
```



```
graph LR; A[Analyse Output format(s)] --> B[Decide on BioXSD format]; B --> C[Write XSD]; C --> D[Create example XML file]; D --> E[Validate]; E --> F[Write parser]; F --> E;
```

14

Thank you for your attention !
Questions?
